User's Manual

Digital Gamma Finder (DGF)

POLARIS

Version 3.40 Doc $Rev:: 13956 $

XIA LLC

31057 Genstar Road
Hayward, CA 94544 USA

Phone: (510) 401-5760; Fax: (510) 401-5761
http://www.xia.com

Disclaimer

Information furnished by XIA is believed to be accurate and reliable. However, XIA assumes no responsibility for its use, or for any infringement of patents, or other rights of third parties, which may result from its use. No license is granted by implication or otherwise under the patent rights of XIA. XIA reserves the right to change the DGF product, its documentation, and the supporting software without prior notice.

Table of Contents

Table of Contents ..... *ii*

1 Overview *1*

1.1 Features **1**

1.2 Specifications **2**

2 Setting up *3*

2.1 Scope of document **3**

2.2 Hardware installation **3**

2.2.1 Host Computer I/O 3

2.2.2 Detector Signal Input 3

2.2.3 Preamplifier power and HV bias 4

2.2.4 AC power 4

2.2.5 Auxiliary signal inputs 4

2.3 Software Installation **4**

3 Polaris Viewer *6*

3.1 Getting Started **6**

3.2 Navigating the Polaris Viewer **8**

3.2.1 Overview 8

3.2.2 System Configuration 9

3.2.3 Acquisition Settings 10

3.2.4 DAQ Run 10

3.2.5 DAQ Data Output 11

3.3 Optimizing Parameters **12**

3.3.1 Noise 12

3.3.2 Energy Filter Parameters 12

3.3.3 Threshold and Trigger Filter Parameters 13

3.3.4 Decay time 13

3.3.5 Dynamic range 13

3.4 Typical Applications **14**

3.4.1 Spectroscopy 14

3.4.2 Spectroscopy with shield 14

3.4.3 Using the Polaris with scintillator detectors 14

3.5 Polaris data structure **15**

3.5.1 IGOR data 15

3.5.2 MCA data files 16

3.5.3 Pulse shape data files 17

3.5.4 Parameter files 18

3.5.5 Information files 18

3.6 User customization **18**

3.6.1 Igor menus and command line 19

3.6.2 Igor procedures 19

3.6.3 DSP customization 19

© XIA LLC 2009. All rights reserved.

4 Hardware description 21
4.1 Analog signal conditioning 21
4.2 Real-time processing unit 21
4.3 Digital signal processor (DSP) 22
4.4 Spectrum Memory 23
4.5 Host interface 23
5 Theory of Operation 24
5.1 Digital Filters for γ-ray detectors 24
5.2 Trapezoidal Filtering in the Polaris 26
5.3 Baselines and preamplifier decay times 27
5.4 Thresholds and Pile-up Inspection 29
5.5 Filter decimation 32
5.6 Count Rates and Livetime 32
6 Appendix 34
6.1 Jumpers 34
6.1.1 Input (JP1, JP9 and JP10) 34
6.1.2 Signal Termination and Attenuation (JP108, JP109, JP112, JP113) 34
6.1.3 Mode (JP103, JP104) 34
6.1.4 VGA (JP106) 35
6.1.5 Compton Veto Polarity (JP110, JP111) 35
6.1.6 HV shutdown (JP20, JP21) 35
6.2 Control and Status Register Bits 36
6.3 Troubleshooting 37
6.3.1 IGOR reports "Function compilation error" at startup 37
6.3.2 IGOR reports missing DLL file 37
6.3.3 USB communication does not work 37
6.3.4 Igor reports missing files at system startup 37
6.3.5 Igor reports FPGA download unsuccessful at system startup 37
6.3.6 Igor can not open files 37
6.3.7 No traces or only flat lines in Oscilloscope 38
6.3.8 Very high input count rate during run 38
6.3.9 Very low livetime during run 38
6.3.10 Large peak at low end of spectrum 38
6.3.11 Spectrum has very wide and blurred peaks 38
6.3.12 Igor reports "need to have at least as many data points as fit parameters" 38

© XIA LLC 2009. All rights reserved.

1 Overview

The Digital Gamma Finder (DGF) family of digital pulse processors features unique capabilities for measuring both the amplitude and shape of pulses in nuclear spectroscopy applications. The DGF architecture was originally developed for use with arrays of multi-segmented HPGe gamma ray detectors, but has since been applied to an ever broadening range of applications.

The DGF Polaris (formerly the Gamma200) is a high-precision, ultra-fast all-digital spectrometer, comprising a single DGF processing channel, a preamplifier power supply and a detector bias supply (up to +/-5,000V) in a compact package. The Polaris provides unparalleled spectral accuracy with up to 64K channels spectrum length, and can on the other hand sustain count rates of up to 750,000 counts per second into the spectrum. Connection to the host computer is by USB or EPP (Extended Parallel Port - IEEE 1284), or an auxiliary 25 pin programmable bidirectional I/O connector for specialty applications.

The Polaris can accept signals from virtually any radiation detector. Signals with decay times as fast as 230ns (from NaI(Tl) for instance) to as slow as 10ms can be processed without the need for external electronics. The Polaris has built-in support for HPGe detectors with a Compton shield: the photomultiplier signal from the shield can be fed directly into the Gate input of the Polaris. No external electronics is necessary. For specialty applications, the Polaris can perform pulse shape analysis, for instance for neutron/gamma discrimination, and can also report data as a list of entries containing energy, time of arrival and even waveforms.

1.1 Features

- Designed for high precision γ-ray spectroscopy with HPGe detectors.
- Directly compatible with scintillator/PMT combinations: NaI, CsI, BGO, and many others.
- Input signal decay time: as fast as 230 ns and up to 10ms, exponentially decaying.
- Wide range of filter rise times: from 50 ns to 45 μs, equivalent to 22 ns to 20 μs shaping times.
- Selectable spectrum length: from 1K to 64K channels, 4.3×10^9 counts per channel.
- Sustained count rate into spectrum: up to 750,000 cps (with scintillator).
- Excellent pile up inspection: double pulse resolution of 100ns.
- Automatic optimization of instrument settings to match detector characteristics.
- Digital oscilloscope and FFT for health-of-system analysis.
- Digital gain stabilization.
- Triggered waveform acquisition for advanced R&D: 14-bit, 40 MSPS, 100 μs. (Contact XIA for 14-bit 65 MSPS and even 80 MSPS option.)
- Compton suppressor input accepts photomultiplier tube input.

© XIA LLC 2009. All rights reserved.

- Includes preamplifier power and high voltage supply.

1.2 Specifications

- ***Inputs (Analog)***

  Signal Input: Selectable input impedance: 50Ω, 90Ω, 250Ω and 10kΩ, ±10V pulsed, ±3V DC. Selectable input attenuation 1:21, 1:12, 1:5 and 1:1.

  Gate Input: (Dual purpose, see below) Input for photomultiplier tube signal from Compton shield. Impedance: 50Ω, ±10V pulsed, ±2V DC.

- ***Inputs (Digital)***

  Gate Input: (Dual purpose, see above) TTL logic input for specialty applications.

  Sync Input: TTL logic input to control time resolved data collection, including scanning and "phase locked loop" applications.

  HV Inhibit: TTL logic input. Selectable logic HI or LO for HV shut down.

- ***Interface***

  USB: Serial interface.

  EPP: Enhanced Parallel Port, IEEE 1284.

  OEM: Auxiliary 25 pin programmable bidirectional I/O connector for specialty applications.

- ***Digital Controls***

  Gain: 80:1 gain range in fine steps.

  Shaping: Digital trapezoidal filter. Rise time and flat top set independently: 0.050 – 45 μs in small steps.

- ***Data Reported***

  Spectrum: 1024-65536 channels, 32-bit deep (4,294,967,295 counts per channel).

  Other: Real time, live time, input and throughput count rates, and Compton shield statistics.

- ***Control I/O (via OEM Port)***

  Control Signals: Sends or receives TTL/CMOS control signals via optional OEM connector, to create flexible custom interfaces to external instruments or industrial equipment. Custom on-board software facilitates integration of the Polaris processor core into dedicated spectroscopy applications.

- ***Other Specifications***

  Detector Supply: High voltage +/- 5000 V, SHV connector, push button on/off, front panel adjust, 60 seconds on/off ramp.

  Preamp Supply: +/- 24 V and +/- 12 V, each rated at 100 mA.

© XIA LLC 2009. All rights reserved.

2 Setting up

2.1 Scope of document

This document covers Polaris devices with serial numbers 100-150.

2.2 Hardware installation

On the front panel of the Polaris spectrometer are controls for detector HV bias as well as the main power switch. All connections are made on the back panel. They include host computer I/O, detector signal input and preamplifier power, detector HV bias, AC power, as well as auxiliary signal inputs.

Some settings need adjustment of internal jumpers, which can be accessed by removing the top cover of the chassis.

2.2.1 Host Computer I/O

Host Computer I/O is made either through the EPP port or the USB port. To use EPP, connect the Polaris EPP port to the host computer's parallel port (printer port). The connection should be made with an IEEE 1284 compliant cable. On the host computer, the BIOS setting for the parallel port has to be EPP, usually the case on modern computers. The EPP address will typically be 0x378 and sometimes 0x278.

To use USB, connect the Polaris USB port to the host computer's USB connector. Make sure the EPP port is disconnected. Whenever you plug in the USB cable or switch on the Polaris, your computer will take a few seconds to recognize the new USB device. Avoid attempts to communicate with the Polaris during that time; it might cause Windows to lock up.

When you connect the Polaris for the first time, Windows will recognize a new device and want to install a driver for an "EZ-USB" controller. The USB driver is located in the *drivers* directory of the software distribution.

2.2.2 Detector Signal Input

The detector signal from the preamplifier connects to the BNC connector labeled "INPUT" on the Polaris back panel. The termination of the signal line can be set to 50Ω, 90Ω or 250Ω, as well as 10kΩ using jumpers on the circuit board (see section 6.1.2 in the appendix). The signal input must fall in the range of ±3V unless jumpers are set for signal attenuation.

© XIA LLC 2009. All rights reserved.

2.2.3 Preamplifier power and HV bias

Preamplifier power (±12V and ±24V) is provided at the DB9 connector labeled "PREAMP POWER". High voltage bias for the detector is provided at the SHV connector labeled ""DETECTOR BIAS". For detectors with thermal shutdown protection, connect the shutdown line to the BNC connector labeled "L/N INHIBIT". Make sure the jumper settings for the shutdown logic matches your particular detector (see section 6.1.6 in the appendix for details).

When the Polaris is switched on, either the "+" or the "-" LED on the front panel is orange, indicating the HV polarity currently set. The polarity can be switched using an internal PCB polarity key: Open the top cover, pull out the small green circuit board near the front right corner, and install it upside down.

The HV bias can be adjusted from 0 to 5000V on the front panel. Use a small screwdriver to turn the potentiometer labeled "ADJUST" to set the voltage. The set voltage is shown in the LCD display. To turn on the high voltage, push the red "ENABLE" button. The polarity LED will change to red, and the LCD display will now show the actual output voltage, ramping up from zero to the set voltage. Pushing the button a second time will ramp down the high voltage back to zero.

2.2.4 AC power

The Polaris can be powered either from 115VAC or 230VAC, depending on the "LINE SELECT" switch. It is rated for 200mA/60Hz (115VAC setting) or 100mA/50Hz (230VAC setting).

2.2.5 Auxiliary signal inputs

The "SYNC" and "GATE" BNC connectors accept auxiliary timing or vetoing signals. Photomultiplier tubes from a Compton rejection shield can be connected to the "GATE" input to veto events from the main detector. No functions are currently implemented for the "SYNC" input. However, it can be customized by XIA through software, for example, to signal the Polaris that a sample is ready, or to advance internal counters.

2.3 Software Installation

The Polaris Viewer, XIA's graphical user interface to set up and run the Polaris, is based on WaveMetrics' IGOR Pro. To run the Polaris Viewer, you have to have IGOR Pro Version 4.0 or higher installed on your computer.

The software resides in a folder *Polaris* with 7 subfolders: *configuration*, *dsp*, *doc*, *drivers*, *firmware*, *MCA* and *pulseshape*. The IGOR control program and the online help

© XIA LLC 2009. All rights reserved.

files are not in any of the subfolders, but are placed one level up in *Polaris*. Make sure you keep this folder organization intact, as the IGOR program and future updates rely on this. Feel free, however, to add folders and subfolders at your convenience.

To install the Polaris Software, run the program Setup.exe from the CD-ROM, and follow the dialog instructions. The setup program will install all necessary drivers for the Polaris. Only if you exit the setup program before the installation is complete will you have to install the following drivers manually:

1. On Windows-98 and later the parallel port can no longer be addressed directly. Even if you use only the USB port for I/O communication with the Polaris, you must run the program "port95nt.exe" from the CD-ROM (located in the *drivers* subdirectory). This will install "DlPortIO" on your computer, a utility to enable direct addressing of the parallel port.

2. If you use the USB port for I/O communication, you have to install the Polaris USB driver on your system. The driver, "xia2k.inf", is located in the *drivers* subdirectory. When Windows detects new hardware, direct it to look for drivers in that folder.

3. Many functions of the Polaris Viewer are precompiled in an Igor .xop file. For Igor to be able to use these functions, the file "Polaris.xop" from the *drivers* directory must be copied into the *Igor Extensions* folder, usually located in C:\Program Files\Wavemetrics\Igor Pro Folder.

© XIA LLC 2009. All rights reserved.

3 Polaris Viewer

3.1 Getting Started

After installing the software and connecting the Polaris to a pulser or detector, double-click on the Polaris.pxp file in the *Polaris* folder to start the Polaris Viewer. When the Viewer has been loaded, it will prompt you to choose the I/O type:

- USB 2.0 using universal serial bus 2.0
- USB 1.1 using universal serial bus 1.1
- EPP using enhanced parallel port
- Offline working without a Polaris spectrometer attached

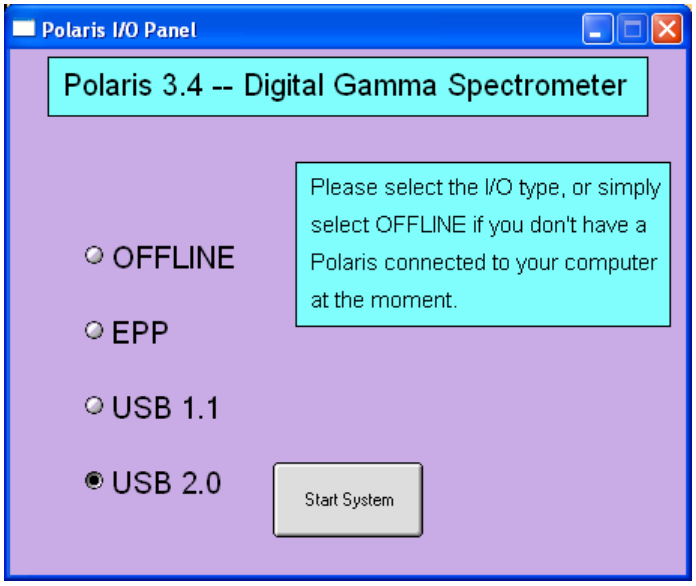
Polaris I/O Panel
Polaris 3.4 -- Digital Gamma Spectrometer
Please select the I/O type, or simply select OFFLINE if you don't have a Polaris connected to your computer at the moment.
OFFLINE
EPP
USB 1.1
USB 2.0
Start System

Figure 3.1 Polaris I/O Panel for choosing host-Polaris I/O type.

If you use the EPP port for I/O communication, set the *EPP Address* to the value of the EPP port on your computer. Typically, the address is 0x378, sometimes 0x278. See section 6.3.5 for details.

Select the I/O type you are using, and then click *Start System*. The I/O light on the Polaris will flash, and some internal relays will click. If no error messages appear, the system is initialized. In the IGOR window you will now see the main Polaris Control Panel from which all work is conducted.

© XIA LLC 2009. All rights reserved.

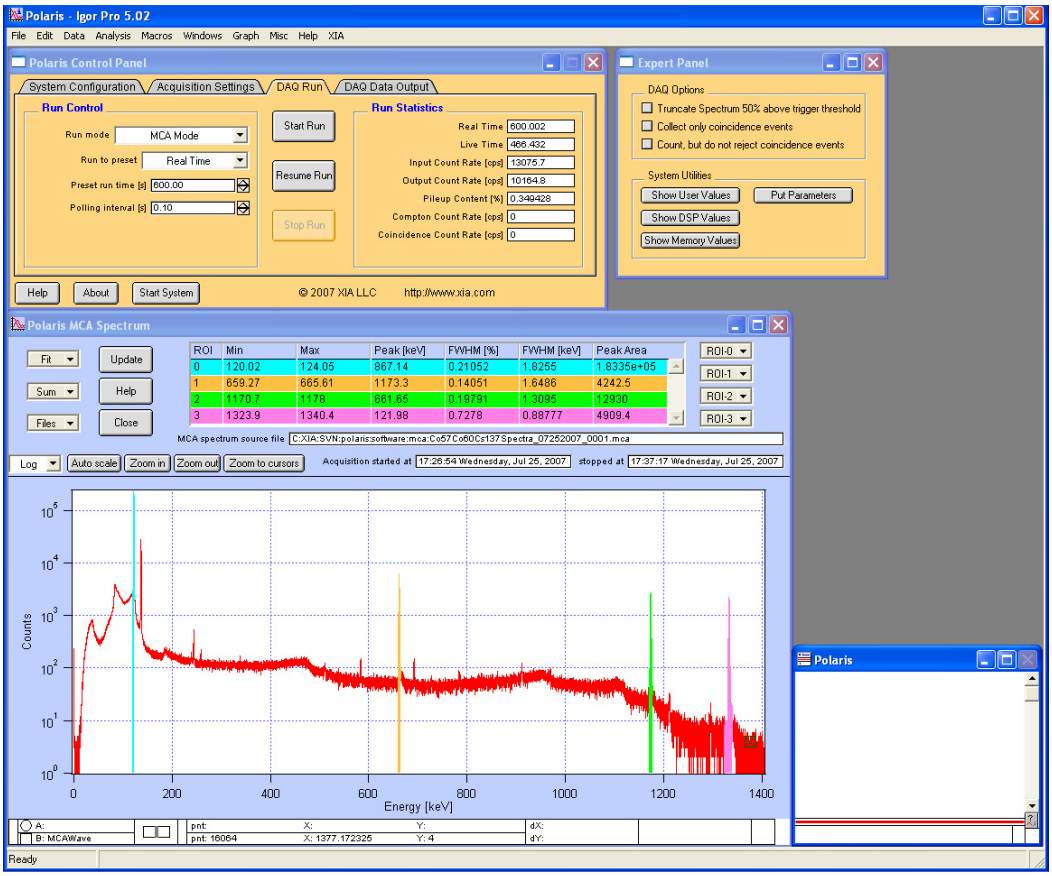

Figure 3.2 The complete Polaris Viewer interface. The panel shown at the top of the left hand side is the Polaris control panel; the graph shown below the Polaris control panel is the MCA spectrum; At the top of the right hand side is the Expert Panel where advanced DAQ options and system utilities are available; the window shown below the Expert Panel is the Polaris Command Line window.

The tabs in the Control Panel are arranged in logical order from left to right. Detailed description of controls and panels can be found in the on-line help from within the Polaris Viewer. To view the help texts, click the *Help* button in the lower left corner of the control panel. In the help topics, click on blue underlined links to jump to cross references. You can also use IGOR's built-in help browser to access the Polaris specific help file by selecting *Help -> Help Topics* from the top menu bar. Choose "Polaris-Help" in the popup menu on the left, and select the appropriate help topic from the list on the right.

For an initial setup and data taking run, you would typically follow the sequence below. Count rates should be kept reasonably low at first, about 5000 cps, especially for determining the decay time. If you encounter problems and strange effects, see the troubleshooting section in the Appendix.

1. In the *System Configuration* tab, make sure the *Polarity* matches the polarity of pulses from your preamplifier.

© XIA LLC 2009. All rights reserved.

2. Click on the *Oscilloscope* button.
   This opens a graph that shows the untriggered signal input. Click *Refresh* to update the display. The pulses should fall between 10% and 80% on the right axis. If no pulses are visible or if they are cut off above 100% or below 0%, click *Adjust* to automatically set the DC offsets. If the pulse amplitude is too large to fall in the display range, increase the *Dynamic Range* in the *Acquisition Settings* tab of the main control panel. Since the offsets might drift, for example after changes in input count rate, it is useful to leave the display open and check the offsets once in a while.

3. In the *System Configuration* tab, enter an estimate of the preamplifier RC decay time, and then click on *Find* to determine the actual decay time.

4. In the *System Configuration* tab, click on *Save Settings to a File* to save the system parameters found so far.

5. Click on the *DAQ Run* tab, set *Run Mode* to MCA Mode, *Run to preset* Real Time, *Preset Run Time* to 30 seconds or so, and then click *Start Run*. During the run, you can click the *Update* button on the *Polaris MCA Spectrum* to view the accumulation of data into the spectrum.

6. After the run is complete, select a known peak from the spectrum and set *Min* and *Max* as the limits of one of the four Regions of Interest (ROI) for a Gauss fit. You can also use the mouse to drag the cursors in the MCA graph to set the limits of the ROI. Click *Fit* to perform the fit. Enter the true energy value in the *Peak [keV]* field to calibrate the energy scale.

7. Click on the *System Configuration* Tab, then again click on *Save Settings to a File*. The Polaris is now set up, and you can take runs and modify parameters to adapt it best to your system.

3.2 Navigating the Polaris Viewer

3.2.1 Overview

The Polaris Viewer consists of a number of graphs and control panels, linked together by the main “Polaris Control Panel”. The “Polaris Control Panel” is divided into 4 tabs, corresponding to the 4 topics summarized below. The *System Configuration* tab contains controls used to initialize the module, and the file and directory settings. The *Acquisition Settings* tab contains controls to adjust parameters such as dynamic range, filter rise time and flat top, and trigger threshold. The *DAQ Run* tab is used to start, resume and stop runs, and in the *DAQ Data Output* tab are controls to set output file directory and name. Below we describe the concepts and principles of using the Polaris Viewer. Detailed information on the individual controls can be found in the online help for each panel.

© XIA LLC 2009. All rights reserved.

3.2.2 System Configuration

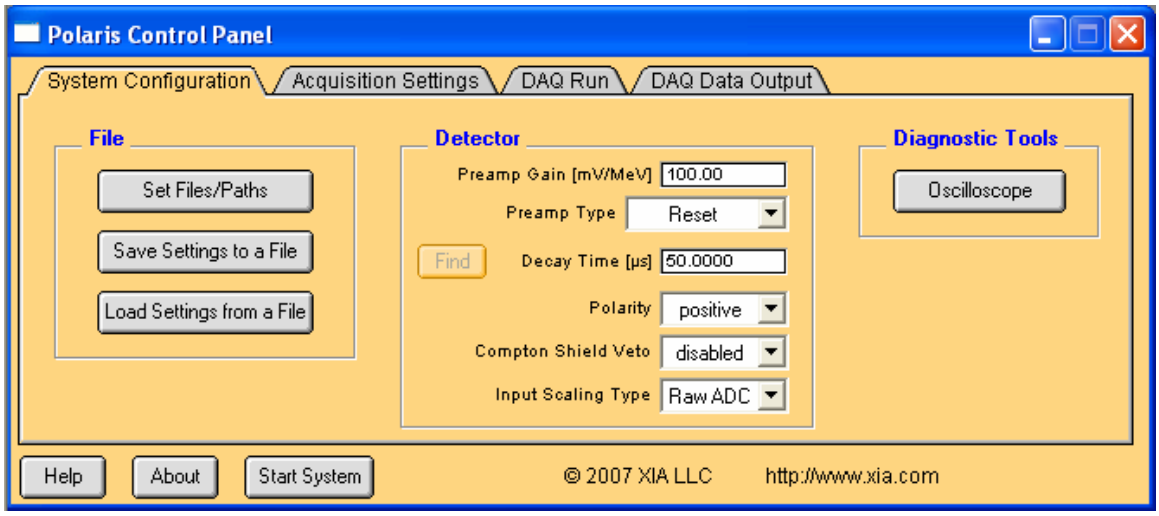

Figure 3.3 Polaris system configuration panel.

The Polaris Viewer comes up in exactly the same state as it was when last saved to file using *File->Save Experiment*. However, the Polaris module itself loses all programming when switched off. When the Polaris is switched on again, only the Host I/O interface is initialized automatically. All the other programmable components need code and configuration files to be downloaded to the module. Clicking the *Start System* button in the *Polaris I/O panel* performs this download.

The Polaris being a digital system, all parameter settings are stored in a settings file. This file is separate from the main IGOR experiment file, to allow saving and restoring different settings for different detectors and applications. Parameter files are saved and loaded with the corresponding buttons in the *System Configuration* tab. At module initialization, the settings are automatically read and applied to the Polaris from the current file which can be changed in the *All Files Panel* (*System Configuration* => *File* => *Set Files/Paths*).

The *System Configuration* tab also has an *Oscilloscope* button linking to a few diagnostic tools. The *Oscilloscope* shows a graph of ADC samples which are untriggered pulses read from the signal input. The time intervals between the samples can be adjusted; for intervals greater than 0.275µs the samples will be averaged over the interval. The main purpose of the Oscilloscope is to make sure that the signal is in range in terms of gain and DC-offset (pulses fall between 10% and 80% on the right axis). The Oscilloscope is also useful to estimate the noise in the system. Clicking on the *FFT Display* button opens the "FFTDisplay", where the noise spectrum can be investigated as a function of frequency. This works best if the Oscilloscope trace contains no pulses, i.e. with the detector attached but no radioactive sources present.

© XIA LLC 2009. All rights reserved.

3.2.3 Acquisition Settings

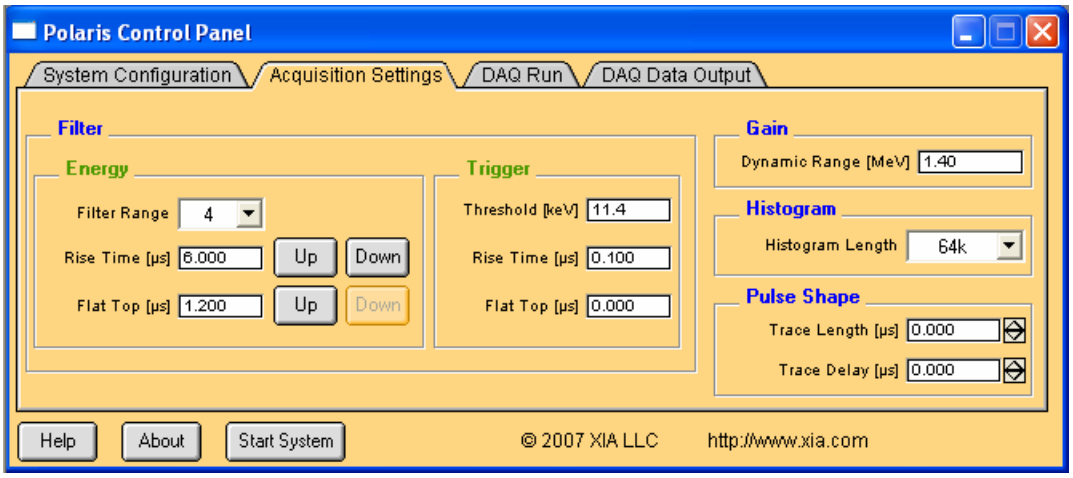

Figure 3.4 Polaris acquisition settings panel.

Internally, the module parameters are handled as binary numbers and bitmasks. The *Acquisition Settings* tab gives access to user parameters in meaningful physical units. Values entered by the user are converted by the Polaris Viewer to the closest value in internal units. You can change rise times of the digital filters, modify the dynamic range, set the trigger threshold, etc. Refer to the online help for detailed descriptions of the parameters.

3.2.4 DAQ Run

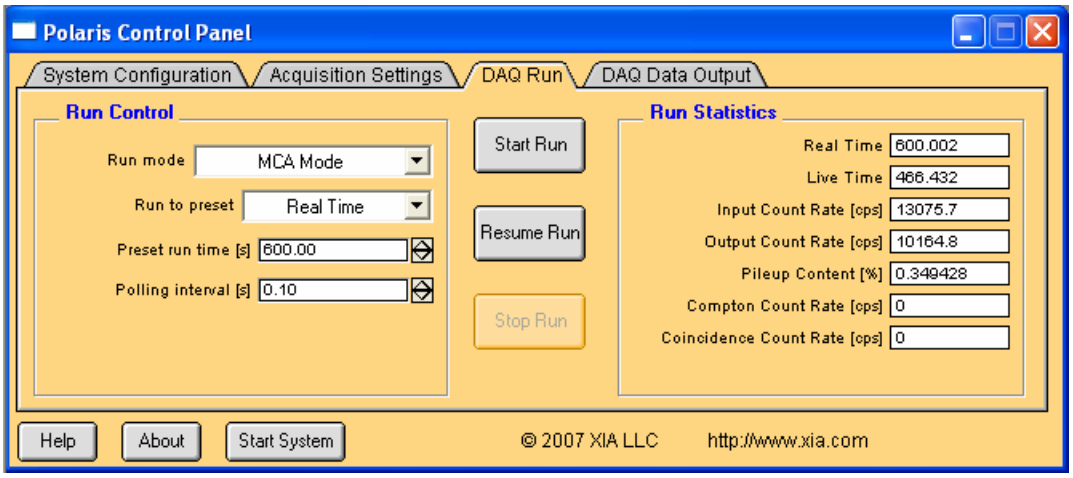

Figure 3.5 Polaris DAQ run panel.

The *DAQ Run* tab is used to start, resume or stop runs. The difference between *Start Run* and *Resume Run* is that for *Start Run*, the run is considered a new run, and the MCA spectrum and run statistics data accumulated during the previous run will be cleared prior

© XIA LLC 2009. All rights reserved.

to the starting of the run, while for *Resume Run* MCA spectrum and run statistics data accumulated during the previous run will not be cleared when resuming the run.

Two run modes are available, one is MCA mode and the other is Pulse Shape (List) mode. In MCA mode, only spectra will be collected. In Pulse Shape (List) mode, you can acquire individual pulses with a time resolution of 25ns. This is a useful tool to find out the characteristics of a given detector and optimize the parameters accordingly. For example, the flat top of the energy filter should ideally be only slightly larger than a typical rise time of a pulse. You can also investigate non-ideal behavior, such as preamplifier overshoots. Pulses are saved to a binary file; see section 3.5.3 for a format description. It is also possible – and has been implemented in other models of XIA's DGF product line – to perform pulse shape analysis in the Polaris during data acquisition and discriminate events accumulated into the spectrum, for example removing events with too long and/or too short rise times. Contact XIA for details.

You can set the run time to either a preset live time or preset real time, or run unlimited. On the right side of the panel, a summary of run statistics is periodically updated during a run, including real time and live time, and various count rates.

After each DAQ run, the spectrum or pulse shape data is saved automatically in the data file specified in the *DAQ Data Output* tab. See section 3.5.2 for the data format of the spectrum file and section 3.5.3 for the data format of the pulse shape data files.

3.2.5 DAQ Data Output

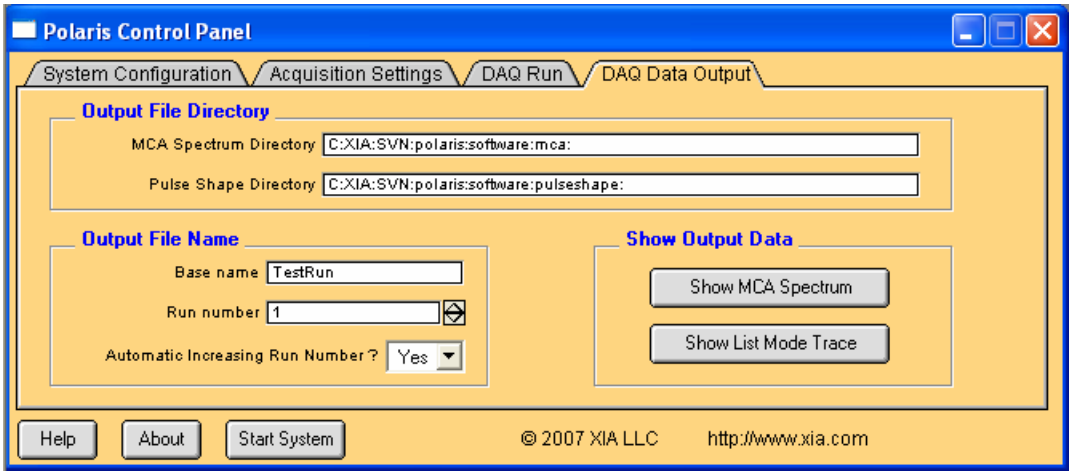

Figure 3.6 Polaris DAQ data output panel.

The *DAQ Data Output* tab is used to set DAQ data output files or paths. You can choose the output file directory for the MCA spectrum or pulse shape data. You can specify the output file name by choosing a base name and run number. The complete output file name will consist of the output file directory plus the output file base name and the run

© XIA LLC 2009. All rights reserved.

number. The run number can be automatically incremented after each DAQ run if that option is enabled.

The *DAQ Data Output* tab also has two buttons (*Show MCA Spectrum* and *Show List Mode Trace*) which can be used to pop up the MCA and list mode data graphs.

3.3 Optimizing Parameters

Optimization of the Polaris's run parameters for best resolution depends on the individual system and usually requires some degree of experimentation. The Polaris Viewer includes several diagnostic tools and settings options to assist the user, as described below.

3.3.1 Noise

For a quick analysis of the electronic noise in the system, you can view a Fourier transform of the incoming signal by selecting *Oscilloscope* → *FFT Display* in the *System Configuration* tab. The graph shows the FFT of the untriggered input sigal of the *Oscilloscope*. By adjusting the "dT" control in the *Oscilloscope* and clicking the *Refresh* button, you can investigate different frequency ranges. For best results, remove any source from the detector and only regard traces without actual events. If you find sharp lines in the 10 kHz to 1 MHz region you may need to find the cause for this and remove it. If you click on the "Filter" button, you can see the effect of the energy filter simulated on the noise spectrum.

3.3.2 Energy Filter Parameters

The main parameter to optimize energy resolution is the rise time of the energy filter. Generally, longer rise times result in better resolution, but reduce the throughput. Optimization should begin with scanning the rise time through the available range. Try 2µs, 4µs, 8µs, 11.2µs, take a run of 60s or so and note changes in energy resolution. Then fine tune the rise time.

The flat top usually needs only small adjustments. For a typical coaxial Ge-detector we suggest to use a flat top of 1.2µs. For a small detector (20% efficiency) a flat top of 0.8µs is a good choice. For larger detectors flat top of 1.2µs and 1.6µs will be more appropriate.

In general the flat top needs to be wide enough to accommodate the longest typical signal rise time from the detector. It then needs to be wider by one filter clock cycle than that minimum, but at least 3 clock cycles. Note that the filter clock cycle ranges from 0.05 to 1.6µs, depending on the filter time range, so that it is not possible to have a very short flat top together with a very long rise time. See the discussion in section 5.5 for further details.

© XIA LLC 2009. All rights reserved.

3.3.3 Threshold and Trigger Filter Parameters

In general, the trigger threshold should be set as low as possible for best resolution. If too low, the input count rate will go up dramatically and "noise peaks" will appear at the minimum and maximum edge of the spectrum. If the threshold is too high, especially at high count rates, low energy events below the threshold can pass the pile-up inspector and pile up with larger events. This increases the measured energy and thus leads to exponential tails on the ideally Gaussian peaks in the spectrum. Ideally, the threshold should be set such that the noise peaks just disappear.

The settings of the trigger filter have only minor effect on the resolution. However, changing the trigger conditions might have some effect on certain undesirable peak shapes. A longer trigger rise time allows the threshold to be lowered more, since the noise is averaged over longer periods. This can help to remove tails on the peaks. A long trigger flat top will help to trigger on slow rising pulses and thus result in a sharper cut off at the threshold.

3.3.4 Decay time

The preamplifier decay time τ is used to correct the energy of a pulse sitting on the falling slope of a previous pulse. The calculations assume a simple exponential decay with one decay constant. A precise value of τ is especially important at high count rates where pulses overlap more frequently. If τ is off the optimum, peaks in the spectrum will broaden, and if τ is very wrong, the spectrum will be significantly blurred.

The first - and usually sufficiently precise - estimate of τ can be obtained from the *Find* routine in the *System Configuration* tab (see item 3 in section 3.1). Measure the decay time several times and settle on the average value.

Fine tuning of τ can be achieved by exploring small variations around the fit value (±2-3%). This is best done at high count rates, as the effect on the resolution is more pronounced. The value of τ found through this way is also valid for low count rates. Manually enter τ in the *System Configuration* tab, take a short run, and note the value of τ that gives the best resolution.

3.3.5 Dynamic range

In most cases, the dynamic range should be set not much larger than the region of interest, for example to 1.5-2.0 MeV for the 1.332 MeV γ-rays of ^{60}Co. This is not a very critical setting, though, since with the 64k channels in the Polaris' MCA, there is still sufficient detail at lower energies even if the dynamic range is set higher than necessary.

© XIA LLC 2009. All rights reserved.

For very high count rates, however, the situation is somewhat different. The architecture of the Polaris is such that the full range of the preamplifier output is mapped to the input range of the ADC, not simply the step height of a single pulse. As a result, at high count rates, when pulses sit on the falling slope of one or even several previous pulses, the dynamic range has to be high enough to accommodate the combined height of the overlapping pulses. For example, if at high count rates up to 3 pulses of 1.332 MeV overlap within the say 50µs decay of the first pulse, the dynamic range is best set to about 3 x 1.332 MeV = 3.99 MeV or higher. Otherwise, the signal will go out of range often, and especially the peaks at the high energy end of the spectrum will lose counts. You can use the Oscilloscope graph to verify if the Dynamic Range is appropriate.

3.4 Typical Applications

In the following section we outline a few typical application examples and give the parameter settings that may be used as a starting point. These example settings are included on the Polaris software distribution.

3.4.1 Spectroscopy

The Polaris is a high-precision, ultra-fast all-digital spectrometer. It provides unparalleled spectral accuracy with up to 64K channels spectrum length, and can on the other hand sustain count rates of up to 750,000 counts per second into the spectrum at an input count rate of over 2.1 million counts per second. The Polaris can accept signals from virtually any radiation detector. Signals with decay times as fast as 230ns (from NaI(Tl) for instance) to as slow as 10ms can be processed without the need for external electronics.

3.4.2 Spectroscopy with shield

In many applications a shielding detector surrounds the sensitive detector. The shield is used to provide a veto when it fires. This helps to reject events in which energy scattered out of the sensitive detector, or background radiation penetrated from the outside. Such a veto signal can be connected to the "Gate" BNC connector on the backside of the Polaris. If the Polaris module comes equipped with the Compton rejection circuitry, the Gate accepts signals directly from a photomultiplier tube. The "Compton Shield Veto" popup menu in the "System Configuration" tab of the Polaris Viewer controls the Gate. When the veto is "disabled", the Gate input is ignored. Otherwise, the event is rejected if a gate pulse is detected within 1 µs of a trigger from the detector input.

3.4.3 Using the Polaris with scintillator detectors

For semiconductor detectors, signals are invariably picked up by a charge-integrating preamplifier with a relatively long decay time (50µs or longer). On the contrary, the light

© XIA LLC 2009. All rights reserved.

signal from scintillators is usually amplified by a photomultiplier (PMT). In this case a preamplifier will most likely not be necessary as the gain of a PMT will almost always be sufficiently high. A second benefit of a charge-integrating preamplifier is that longer filters can be used on its step-like output to suppress the electronic noise and improve energy resolution. In scintillator applications, however, the energy resolution is rarely limited by the electronic noise. Hence, we can take advantage of the often fairly short decay time constants for the scintillation light output in order to achieve high count rates, and at the same time simplify the system. The current output from the PMT traces the scintillation light intensity and can be fed directly to the Polaris inputs. If the PMT is operated at negative high voltage, its anode is at ground potential and we can pick off the current directly. If the PMT is powered by positive high voltage, its anode is at high potential and the current has to be picked off through a coupling capacitor. In order to avoid the introduction of unwanted time constants, it is advisable to couple the anode current capacitively into a current-to-voltage converting preamplifier. Some manufacturers sell PMTs that are powered with positive high voltage with a base that includes an integrating preamplifier. This preamp can be converted into a current-to-voltage converter by removing its integrating capacitor. It may also be necessary to improve the local RC-filtering of high voltage inside the PMT base. With these modifications the preamplifier output will trace the scintillation light in time, and its integral will be proportional to the energy deposited in the scintillator.

3.5 Polaris data structure

3.5.1 IGOR data

In the Polaris viewer, a number of output variables contain data that might be useful for calculations and/or custom displays. They are listed in Table 3.1.

IGOR variable or wave name	Description
root:polaris:LiveTime	Polaris live time in sec.
root:polaris:RunTime	Polaris run time in sec.
root:polaris:InputCountRate	Input count rate in cps
root:polaris:OutputCountRate	Output count rate in cps
root:polaris:ShieldCountRate	Count rate at “Gate” input
root:polaris:ComptonCountRate	Coincidence rate of detector and “Gate” pulses
root:polaris:MCAwave	MCA spectrum wave

Table 3.1**: IGOR output variables**

The input variables shown below should only be changed in the control panel to make sure all dependencies are updated properly.

IGOR variable or wave name	Description
root:polaris:DynamicRange	Dynamic range in MeV
root:polaris:PreampGain	Preamplifier gain in mV/MeV
root:polaris:TriggerThreshold	Trigger threshold in keV

© XIA LLC 2009. All rights reserved.

root:polaris:BaselinePercent	Default offset level in %
root:polaris:DetectorTau	Preamplifier decay time
root:polaris:HistogramLength	Histogram length (number of bins)
root:polaris:TraceLength	Trace length of pulse shape data
root:polaris:TraceDelay	Pre-trigger time of pulse shape data
root:polaris:XDT	Time step of oscilloscope trace
root:polaris:TriggerPeakingTime	Rise time of trigger filter
root:polaris:TriggerGapTime	Flat top time of trigger filter
root:polaris:EnergyPeakingTime	Rise time of energy filter
root:polaris:EnergyGapTime	Flat top time of energy filter
root:polaris:PresetRunTime	Preset run time
root:polaris:PresetRunType	0-infinite, 1-preset real time, 2- preset live time
root:polaris:RunTimeUnit	Time multiplier: 60 for min, 3600 for hours, etc

Table 3.2**: Igor Input Variables**

3.5.2 MCA data files

MCA files are saved automatically after each run to the filename specified in the *DAQ Data Output* tab as a binary (unsigned 32-bit integer words) file (.mca file). Additionally on the *MCA Spectrum Display* graph, MCA data can also be exported to an IGOR text file (.itx) in ASCII format as shown in the example below, or to an ORTEC® CHN file. In the header of the .itx file, the most important operating conditions are summarized. The user is prompted for entries to the "Detector", "Condition", and "Operator" keywords before saving the spectrum. The header is followed by the MCA data (each line has the number of counts in one channel). The last line of the file contains an IGOR command for scaling the MCA in the same energy scale as originally saved.

```
IGOR
X // XIA Polaris MCA data saved Fri, Apr 12, 2002, 6:19:21 PM
X // Detector =
X // Condition =
X // Operator =
X // Run Time [sec]= 6360.92
X // Live Time [sec]= 5224.5
X // Rise Time [us] = 6
X // Flat Top [us] = 1.2
X // Decay Time [us] = 47.35
X // Dynamic Range [MeV] = 2.24534
X // Trigger Threshold [keV] = 15.1876
X // Input Count Rate [cps] = 9417.14
X // Output Count Rate [cps] = 7734.71

WAVES          MCAch0
BEGIN
0
0
0
```


© XIA LLC 2009. All rights reserved.

```
0
0
…
0
0
END
X SetScale/P x 0,0.0379690920408061,"", MCAch0; SetScale y 0,0,"", MCAch0
```

3.5.3 Pulse shape data files

Pulse shape data is saved in binary form (unsigned 16-bit integer) and the binary file has the extension of ".bin". One or more readouts of the Polaris' output buffer are saved or appended in a single file. The buffer length is 8K words; each word is a 16-bit unsigned integer. A parameter file with the same name, but extension ".set" is saved together with the data file.

The output data can be written in a number of formats, though currently only one format is actually used. The Polaris Viewer has built in functions to parse the files and display event data and waveforms. If user code is used to read the files, it should access the three variables BUFHEADLEN, EVENTHEADLEN, and CHANHEADLEN in the parameter file of a particular run to navigate through the data set.

The buffer content always starts with a buffer header of length BUFHEADLEN. Currently, BUFHEADLEN is six, and the six words are:

Word #	Variable	Description
0	BUF_NDATA	Number of words in this buffer
1	BUF_MODNUM	Module number
2	BUF_FORMAT	Format descriptor
3	BUF_TIMEHI	Run start time, high word
4	BUF_TIMEMI	Run start time, middle word
5	BUF_TIMELO	Run start time, low word

Table 3.3**: Buffer header data format.**

Following the buffer header, the events are stored in sequential order. Each event starts out with an event header of length EVENTHEADLEN. Currently, EVENTHEADLEN=3, and the three words are:

Word #	Variable	Description
0	EVT_PATTERN	Hit pattern
1	EVT_TIMEHI	Event time, high word
2	EVT_TIMELO	Event time, low word

Table 3.4**: Event header data format.**

© XIA LLC 2009. All rights reserved.

The LSB (bit 0) of the hit pattern, if set, indicates that the channel has recorded an event. The other bits are unused and reserved. After the event header follows the channel information: A channel header of length CHANHEADLEN, which may be followed by waveform data. For standard List Mode, the only pulse shape data format currently supported, CHANHEADLEN=9, and the nine words are

Word #	Variable	Description
0	CHAN_NDATA	Number of words for this channel
1	CHAN_TRIGTIME	Fast trigger time
2	CHAN_ENERGY	Energy
3	CHAN_XIAPSA	XIA PSA value
4	CHAN_USERPSA	User PSA value
5	Reserved	Raw data
6	Reserved	Raw data
7	Reserved	Raw data
8	Reserved	Raw data

Table 3.5**: Channel header, possibly followed by waveform data. If CHAN_NDATA>9 there will be waveform data following this channel header.**

Any waveform data for this channel would then follow this header. An offline analysis program can recognize this by computing N_WAVE_DATA=CHAN_DATA-9. If N__WAVE_DATA is greater than zero, it indicates the number of waveform data words to follow.

3.5.4 Parameter files

Polaris Parameter files are saved as a list of 416 numbers in binary form (signed 16-bit words) with a file extension of ".set". The numbers correspond to a list of 416 DSP variable names stored in "Polaris.var", an ASCII file in the *dsp* folder. As the DSP variables might change and shift, it is important to always refer to the ASCII file to match a variable with its value. The DSP variable values are downloaded directly to the Polaris' DSP, and converted into user values in the Polaris viewer.

3.5.5 Information files

The Polaris information files have an extension of ".ifm" and are saved automatically at the end of each DAQ run. It records the time for the Start and Stop of the DAQ run.

3.6 User customization

© XIA LLC 2009. All rights reserved.

The Polaris Viewer provides all necessary functions to set up and run the Polaris, and a set of basic analysis tools. However, the user might be interested in using the numerous tools and functions available in IGOR to perform custom curve fits, calculate results or even perform macros or scripts for routine tasks.

3.6.1 Igor menus and command line

Without giving a full introduction into IGOR, which can be found in the WaveMetrics documentation, we list a few useful tools and features below:

- Help for IGOR is available through IGOR's help browser, located under *Help* in the top menu bar.

- Tools to modify graphs are available from the *Graph* menu in the top menu bar if the graph is the front window. You can modify symbols, trace appearance, axes, etc. Most items (axes, traces, labels etc) in a graph can also be modified by double-clicking on the item. Useful keyboard shortcuts are Ctrl-I to show or hide cursors on a graph, and Ctrl-A to rescale a graph to the full size.

- The full range of IGOR's analysis tools are available in the *Analysis* menu in the top menu bar. This includes curve fits, wave statistics, and various smoothing functions. Curve fits can be customized with user defined fit functions. Note that most Polaris data resides in the polaris subfolder, and have to be addressed as "root:polaris:MCAwave" rather than simply "MCAwave" (see section 3.5.1 for details).

- Every IGOR Pro experiment file has a history window with command line for entering and logging commands and messages. If you modify graphs or panels, the modification commands are usually printed in the history window, from where they can be copied to the command line (to edit and/or repeat) or into a user procedure. The command line is also useful to issue commands such as to "duplicate" the current spectrum to compare it with other spectra from file or from subsequent runs, or for simple calculations.

3.6.2 Igor procedures

All underlying functions and procedures of the Polaris Viewer are available in IGOR's procedure windows, which can be accessed by clicking Windows on the top of the Igor window and then selecting Other Windows.

3.6.3 DSP customization

© XIA LLC 2009. All rights reserved.

For demanding applications, pulse shape analysis can be performed in the Polaris digital signal processor while data is taken. Examples include rejecting events based on certain user defined criteria, or calculating timing or energy quantities from the acquired waveforms. The DSP code is set up with calls to "user" routines, which can be modified by users and compiled into the main code. Contact XIA for details.

© XIA LLC 2009. All rights reserved.

4 Hardware description

The Polaris is a single channel unit designed for Gamma-ray spectroscopy and waveform capturing. It incorporates five functionally different building blocks, which we describe below. This section concentrates on the functionality aspect. Technical specification can be found in section 1.2.

4.1 Analog signal conditioning

Each analog input is first fed into a signal conditioning unit. The task of this circuitry is to adapt the incoming signal to the input voltage range of the ADC, which spans 1.00V. The input signal is adjusted for offset, and there is a computer-controlled gain stage. This helps to bring the signal into the ADC's voltage range and set the dynamic range of the channel.

The ADC is not a peak sensing ADC, but acts as a waveform digitizer. In order to avoid aliasing, we remove the high frequency components from the incoming signal prior to feeding it into the ADC. The anti-aliasing filter, a 3rd order active Sallen-Key filter, cuts off sharply at the Nyquist frequency, namely half the ADC sampling frequency.

Though the Polaris can work with many different signal forms, best performance is to be expected when sending the output from a charge integrating preamplifier directly to the Polaris without any further shaping.

4.2 Real-time processing unit

The real time processing unit consists of a field programmable gate array (FPGA) and a FIFO memory. The data stream from the ADCs is sent to this unit at the full ADC sampling rate. Using a pipelined architecture, the signals are also processed at this high rate, without the help of the on-board digital signal processor (DSP).

The real-time processing unit (RTPU) applies digital filtering to perform essentially the same action as a shaping amplifier. The important difference is in the type of filter used. In a digital application is easy to implement finite impulse response filters, and we use a trapezoidal filter. The flat top will typically cover the rise time of the incoming signal and makes the pulse height measurement less sensitive to variations of the signal shape.

Secondly, the RTPU contains a pileup inspector. This logic ensures that if a second pulse is detected too soon after the first, so that it would corrupt the first pulse height measurement, both pulse are rejected as piled up. The pileup inspector is, however, not very effective in detecting pulse pileup on the rising edge of the first pulse, i.e. in general pulses must be separated by their rise time to be effectively recognized as different

© XIA LLC 2009. All rights reserved.

pulses. Therefore, for high count rate applications, the pulse rise times should be as short as possible, to minimize the occurrence of pileup peaks in the resulting spectra.

If a pulse was detected and passed the pileup inspector, a trigger may be issued. That trigger would notify the DSP that there are raw data available now. If a trigger was issued the data remain latched until the RTPU has been serviced by the DSP.

The third component of the RTPU is a FIFO memory, which is controlled by the pile up inspector logic. The FIFO memory is continuously being filled with waveform data from the ADC. On a trigger it is stopped, and the read pointer is positioned such that it points to the beginning of the pulse that caused the trigger. When the DSP collects event data, it can read any fraction of the stored waveform, up to the full length of the FIFO.

4.3 Digital signal processor (DSP)

The DSP controls the operation of the Polaris, reads raw data from the RTPU, reconstructs true pulse heights, applies time stamps, and prepares data for output to the host computer, and increments spectra in the external memory.

The host computer communicates with the board, via the EPP or USB interface, using a direct memory access (DMA) channel. Reading and writing data to DSP memory does not interrupt its operation, and can occur even while a measurement is underway. Note that EPP transfers introduce additional noise to the signal, so it is best to avoid transfers while a run is in progress. USB transfers do not show this problem.

The host sets variables in the DSP memory and then calls DSP functions to program the hardware. Through this mechanism all gain and offset DACs are set and the RTPU are programmed.

The RTPU processes its data without support from the DSP, once it has been set up. When it generates a trigger, an interrupt request is sent to the DSP. The DSP responds with reading the required data from the RTPU and storing those in memory. It then returns from the interrupt routine without processing the data to minimize the DSP induced dead time. The event processing routine works from the data in memory to generate the requested output data.

In this scheme, the greatest processing power is located in the RTPU. Implemented in a FPGA, it processes the incoming waveforms from its associated ADC in real time and produces, for each valid a event, a small set of distilled data from which pulse heights and arrival times can be reconstructed. The computational load for the DSP is much reduced, as it has to react only on an event-by-event basis and has to work with only a small set of numbers for each event.

© XIA LLC 2009. All rights reserved.

4.4 Spectrum Memory

Energy spectra are accumulated in a 64k x 32bit memory chip, allowing for 64k bins with more than 4 billion counts each. The DSP passes energy values to a memory manager implemented in an FPGA, which then increments the corresponding bin in the spectrum. The host computer can read the spectrum via the DMA bus, without interrupting the DSP operation. This architecture further reduces the computational load for the DSP and allows for fast transfers of spectrum data.

For special applications, for example accumulating several independent spectra in a single run, this memory can be extended up to a total of 512k bins. Contact XIA for details.

4.5 Host interface

The EPP interface through which the host communicates with the Polaris is implemented in its own FPGA. The configuration of this gate array is stored in a PROM, which is placed in the only DIP-8 IC-socket on the Polaris board.

The USB interface is implemented in a separate microcontroller chip. It is configured by a separate on-board PROM. The USB microcontroller also reads temperature data from an on-board thermometer, which the DSP can use to detect and compensate gain drifts.

© XIA LLC 2009. All rights reserved.

5 Theory of Operation

5.1 Digital Filters for γ-ray detectors

Energy dispersive detectors, which include such solid state detectors as Si(Li), HPGe, HgI_2, CdTe and CZT detectors, are generally operated with charge sensitive preamplifiers as shown in Figure 5.1a. Here the detector D is biased by voltage source V and connected to the input of preamplifier A which has feedback capacitor C_f and feedback resistor R_f.

The output of the preamplifier following the absorption of an γ-ray of energy E_x in detector D is shown in Figure 5.1b as a step of amplitude V_x (on a longer time scale, the step will decay exponentially back to the baseline, see section 5.3). When the γ-ray is absorbed in the detector material it releases an electric charge $Q_x = E_x/\varepsilon$, where ε is a material constant. Q_x is integrated onto C_f, to produce the voltage $V_x = Q_x/C_f = E_x/(\varepsilon C_f)$. Measuring the energy E_x of the γ-ray therefore requires a measurement of the voltage step V_x in the presence of the amplifier noise σ, as indicated in Figure 5.1b.

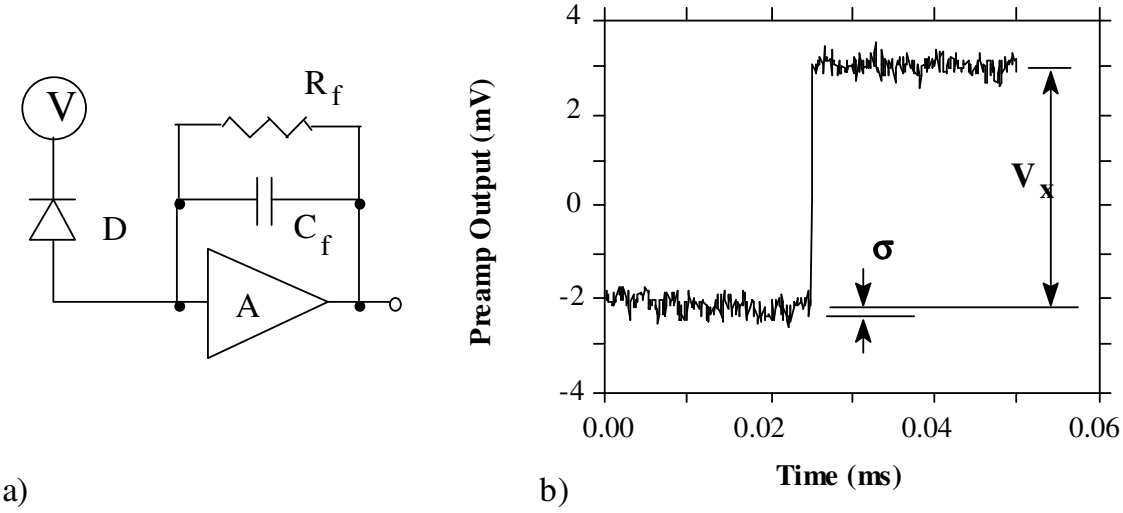

Figure 5.1 a) Charge sensitive preamplifier with RC feedback; b) Output on absorption of an γ-ray.

Reducing noise in an electrical measurement is accomplished by filtering. Traditional analog filters use combinations of a differentiation stage and multiple integration stages to convert the preamp output steps, such as shown in Figure 5.1b, into either triangular or semi-Gaussian pulses whose amplitudes (with respect to their baselines) are then proportional to V_x and thus to the γ-ray's energy.

Digital filtering proceeds from a slightly different perspective. Here the signal has been digitized and is no longer continuous, but is instead a string of discrete values, such as

© XIA LLC 2009. All rights reserved.

shown in Figure 5.2. Figure 5.2 is actually just a subset of Figure 5.1b, which was digitized by a Tektronix 544 TDS digital oscilloscope at 10 MSA (megasamples/sec). Given this data set, and some kind of arithmetic processor, the obvious approach to determining V_x is to take some sort of average over the points before the step and subtract it from the value of the average over the points after the step. That is, as shown in Figure 5.2, averages are computed over the two regions marked "Length" (the "Gap" region is omitted because the signal is changing rapidly here), and their difference taken as a measure of V_x. Thus the value V_x may be found from the equation:

$$V_{x,k} = -\sum_{i(before)} W_i V_i + \sum_{i(after)} W_i V_i \qquad (5.1)$$

where the values of the weighting constants W_i determine the type of average being computed. The sums of the values of the two sets of weights must be individually normalized.

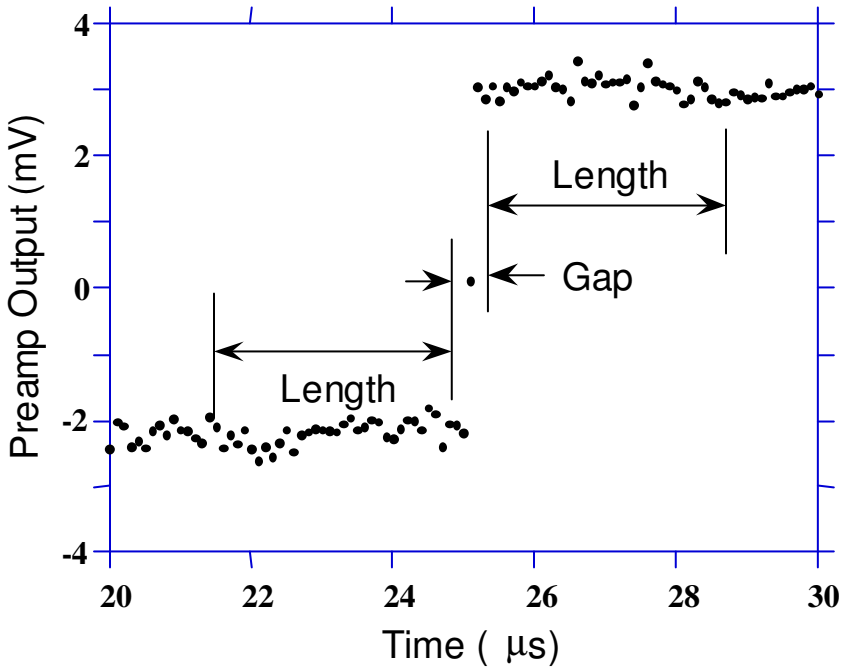

Figure 5.2 Digitized version of the data of **Figure 5.1** b) in the step region.

The primary differences between different digital signal processors lie in two areas: what set of weights $\{w_i\}$ is used and how the regions are selected for the computation of Eqn. 5.1. Thus, for example, when the weighting values decrease with separation from the step, then Eqn. 5.1 produces "cusp-like" filters. When the weighting values are constant, one obtains triangular (if the gap is zero) or trapezoidal filters. The concept behind cusp-like filters is that, since the points nearest the step carry the most information about its height, they should be most strongly weighted in the averaging process. How one chooses the filter lengths results in time variant (the lengths vary from pulse to pulse) or

© XIA LLC 2009. All rights reserved.

time invariant (the lengths are the same for all pulses) filters. Traditional analog filters are time invariant. The concept behind time variant filters is that, since the γ-rays arrive randomly and the lengths between them vary accordingly, one can make maximum use of the available information by setting Length to the interpulse spacing.

In principal, the very best filtering is accomplished by using cusp-like weights and time variant filter length selection. There are serious costs associated with this approach however, both in terms of computational power required to evaluate the sums in real time and in the complexity of the electronics required to generate (usually from stored coefficients) normalized $\{w_i\}$ sets on a pulse by pulse basis.

The Polaris takes a different approach because it was optimized for very high speed operation. It implements a fixed length filter with all w_i values equal to unity and in fact computes this sum afresh for each new signal value k. Thus the equation implemented is:

$$LV_{x,k} = -\sum_{i=k-2L-G+1}^{k-L-G} V_i + \sum_{i=k-L+1}^{k} V_i \qquad (5.2)$$

where the filter length is L and the gap is G. The factor L multiplying $V_{x,k}$ arises because the sum of the weights here is not normalized. Accommodating this factor is trivial.

While this relationship is very simple, it is still very effective. In the first place, this is the digital equivalent of triangular (or trapezoidal if $G \neq 0$) filtering which is the analog industry's standard for high rate processing. In the second place, one can show theoretically that if the noise in the signal is white (i.e. Gaussian distributed) above and below the step, which is typically the case for the short shaping times used for high signal rate processing, then the average in Eqn. 5.2 actually gives the best estimate of V_x in the least squares sense. This, of course, is why triangular filtering has been preferred at high rates. Triangular filtering with time variant filter lengths can, in principle, achieve both somewhat superior resolution and higher throughputs but comes at the cost of a significantly more complex circuit and a rate dependent resolution, which is unacceptable for many types of precise analysis. In practice, XIA's design has been found to duplicate the energy resolution of the best analog shapers while approximately doubling their throughput, providing experimental confirmation of the validity of the approach.

5.2 Trapezoidal Filtering in the Polaris

From this point onward, we will only consider trapezoidal filtering as it is implemented in the Polaris according to Eqn. 5.2. The result of applying such a filter with Length L=1μs and Gap G=0.4μs to a γ-ray event is shown in Figure 5.3. The filter output is clearly trapezoidal in shape and has a risetime equal to L, a flattop equal to G, and a symmetrical falltime equal to L. The basewidth, which is a first-order measure of the filter's noise reduction properties, is thus 2L+G.

© XIA LLC 2009. All rights reserved.

This raises several important points in comparing the noise performance of the Polaris to analog filtering amplifiers. First, semi-Gaussian filters are usually specified by a *shaping time*. Their rise time is typically twice this and their pulses are not symmetric so that the basewidth is about 5.6 times the shaping time or 2.8 times their rise time. Thus a semi-Gaussian filter typically has a slightly better energy resolution than a triangular filter of the same rise time because it has a longer filtering time. This is typically accommodated in amplifiers offering both triangular and semi-Gaussian filtering by stretching the triangular rise time a bit, so that the *true* triangular rise time is typically 1.2 times the selected semi-Gaussian rise time. This also leads to an apparent advantage for the analog system when its energy resolution is compared to a digital system with the same nominal rise time.

One important characteristic of a digitally shaped trapezoidal pulse is its extremely sharp termination on completion of the basewidth 2L+G. This may be compared to analog filtered pulses which have tails which may persist up to 40% of the rise time, a phenomenon due to the finite bandwidth of the analog filter. As we shall see below, this sharp termination gives the digital filter a definite rate advantage in pileup free throughput.

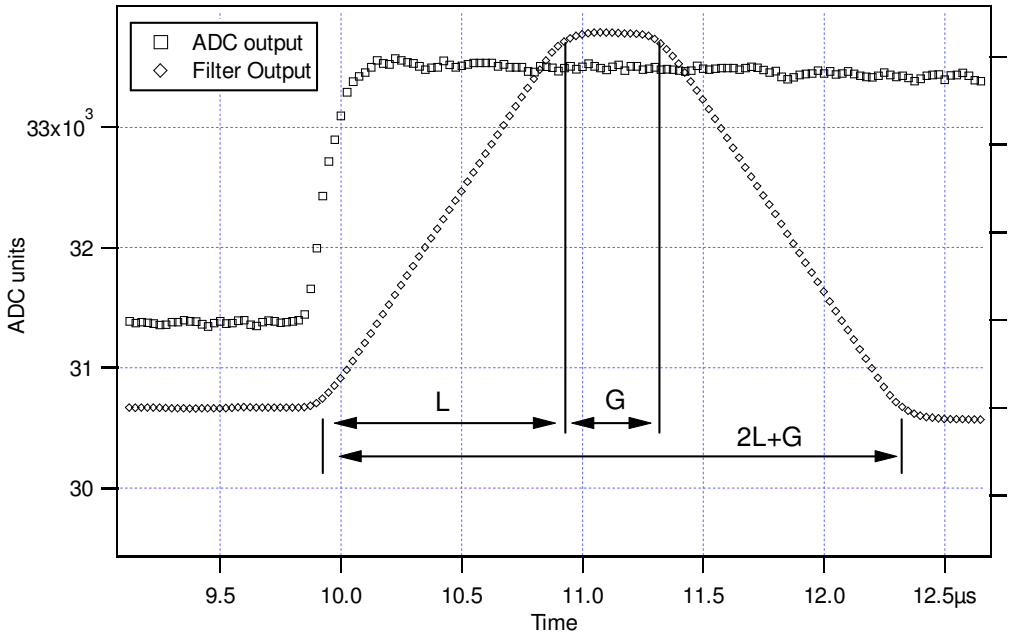

Figure 5.3: Trapezoidal filtering of a preamp step with L=1µs and G=0.4µs

5.3 Baselines and preamplifier decay times

Figure 5.4 shows an event over a longer time interval to show how the filter treats the preamplifier noise in regions when no γ-ray pulses are present. As may be seen the effect of the filter is both to reduce the amplitude of the fluctuations and reduce their high

© XIA LLC 2009. All rights reserved.

frequency content. This signal is termed the *baseline* because it establishes the reference level from which the γ-ray peak amplitude V_x is to be measured. The mean value of the baseline is zero if no pulses are present. The fluctuations in the baseline have a standard deviation σ_e which is referred to as the *electronic noise* of the system, a number which depends on the rise time of the filter used. Riding on top of this noise, the γ-ray peaks contribute an additional noise term, the *Fano noise*, which arises from statistical fluctuations in the amount of charge Q_x produced when the γ-ray is absorbed in the detector. This Fano noise σ_f adds in quadrature with the electronic noise, so that the total noise σ_t in measuring V_x is found from

$$\sigma_t = \text{sqrt}(\sigma_f^2 + \sigma_e^2) \tag{5.3}$$

The Fano noise is only a property of the detector material. The electronic noise, on the other hand, may have contributions from both the preamplifier and the amplifier. When the preamplifier and amplifier are both well designed and well matched, however, the amplifier's noise contribution should be essentially negligible. Achieving this in the mixed analog-digital environment of a digital pulse processor is a non-trivial task, however.

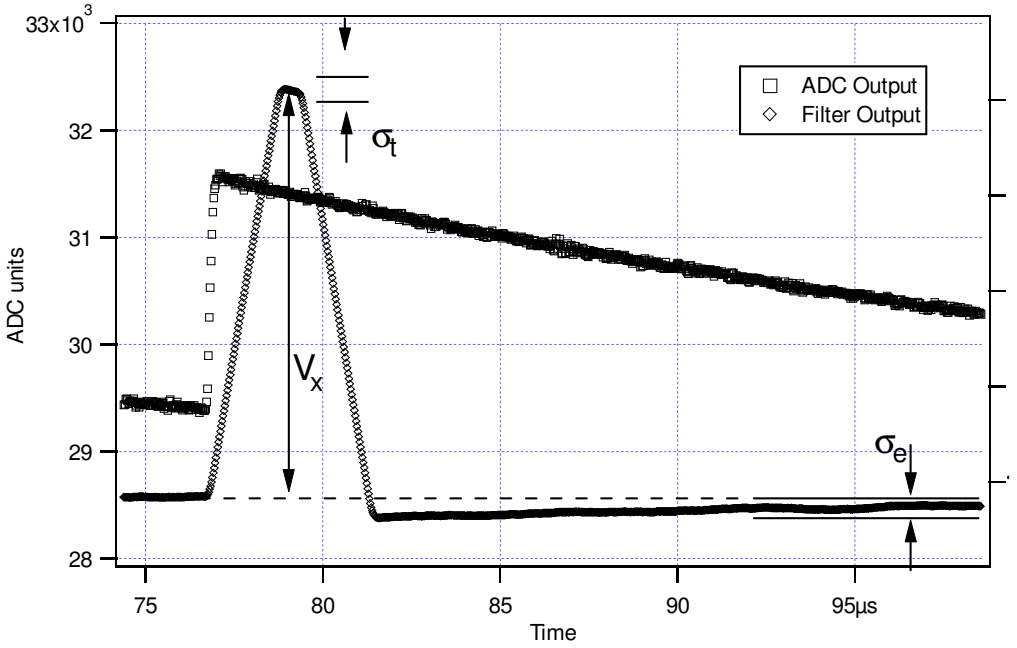

Figure 5.4: Aγ-ray event displayed over a longer time period to show baseline noise and the effect of preamplifier decay time.

In the general case, however, the mean baseline value is not zero. This situation arises whenever the slope of the preamplifier signal it not zero between γ-ray pulses. This can be seen from Eqn. 5.2. When the slope is not zero, the mean values of the two sums will differ because they are taken over regions separated in time by L+G, on average.

© XIA LLC 2009. All rights reserved.

With a RC-type preamplifier, since pulses are not the simple step functions described above, the slope of the preamplifier is rarely zero. Every step decays exponentially back to the DC level of the preamplifier. During such a decay, the baselines are obviously not zero. This can be seen in Figure 5.4, where the filter output during the exponential decay after the pulse is below the initial level. Note also that the flat top region is sloped downwards.

Using the decay constant τ, the baselines can be mapped back to the DC level. This allows precise determination of γ-ray energies, even if the pulse sits on the falling slope of a previous pulse. The value of τ, being a characteristic of the preamplifier, has to be determined by the user and host software and downloaded to the module.

5.4 Thresholds and Pile-up Inspection

As noted above, we wish to capture a value of V_x for each γ-ray detected and use these values to construct a spectrum. This process is also significantly different between digital and analog systems. In the analog system the peak value must be "captured" into an analog storage device, usually a capacitor, and "held" until it is digitized. Then the digital value is used to update a memory location to build the desired spectrum. During this analog to digital conversion process the system is dead to other events, which can severely reduce system throughput. Even single channel analyzer systems introduce significant deadtime at this stage since they must wait some period (typically a few microseconds) to determine whether or not the window condition is satisfied.

Digital systems are much more efficient in this regard, since the values output by the filter are already digital values. All that is required is to take the filter sums, reconstruct the energy V_x, and add it to the spectrum. In the Polaris, the filter sums are continuously updated by the RTPU (see section 4.2), and only have to be read out by the DSP when an event occurs. Reconstructing the energy and incrementing the spectrum is done by the DSP, so that the RTPU is ready to take new data immediately after the readout. This usually takes much less than one rise time, so that no system deadtime is produced by a "capture and store" operation. This is a significant source of the enhanced throughput found in digital systems.

© XIA LLC 2009. All rights reserved.

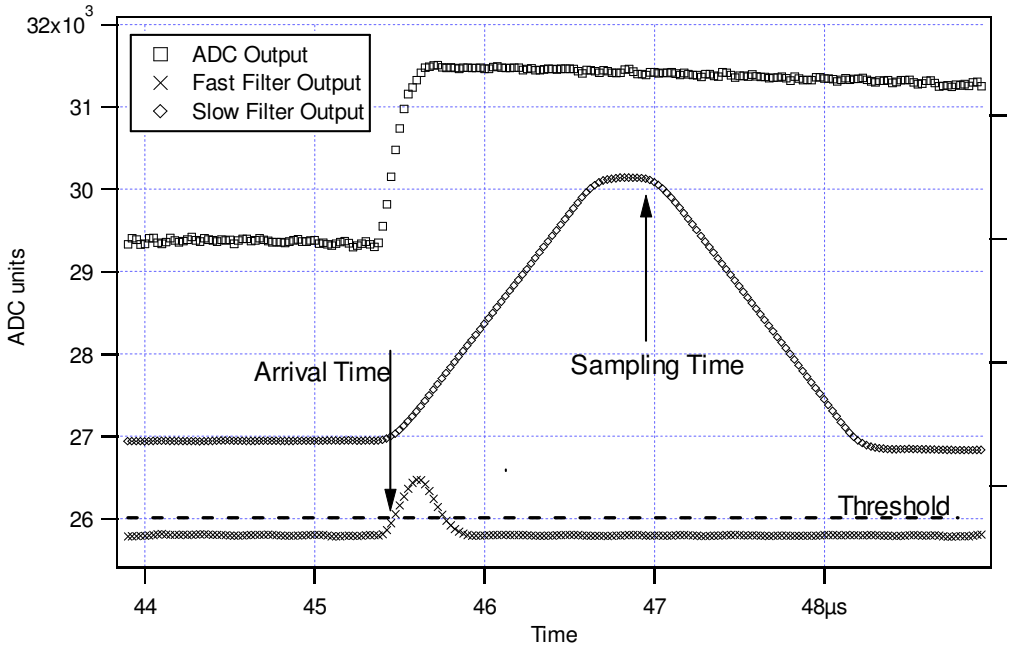

Figure 5.5: Peak detection and sampling in the Polaris.

The peak detection and sampling in the Polaris is handled as indicated in Figure 5.5. Two trapezoidal filters are implemented, a *fast filter* and a *slow filter*. The fast filter is used to detect the arrival of γ-rays, the slow filter is used for the measurement of V_x, with reduced noise at longer rise times. The fast filter has a filter length $L_f = 0.1\mu s$ and a gap $G_f = 0.1\mu s$. The slow filter has $L_s = 1.2\mu s$ and $G_s = 0.35\mu s$.

The arrival of the γ-ray step (in the preamp output) is detected by digitally comparing the fast filter output to THRESHOLD, a digital constant set by the user. Crossing the threshold starts a counter to count PEAKSAMP clock cycles to arrive at the appropriate time to sample the value of the slow filter. Because the digital filtering processes are deterministic, PEAKSAMP depends only on the values of the fast and slow filter constants and the risetime of the preamplifier pulses. The slow filter value captured following PEAKSAMP is then the slow digital filter's estimate of V_x.

© XIA LLC 2009. All rights reserved.

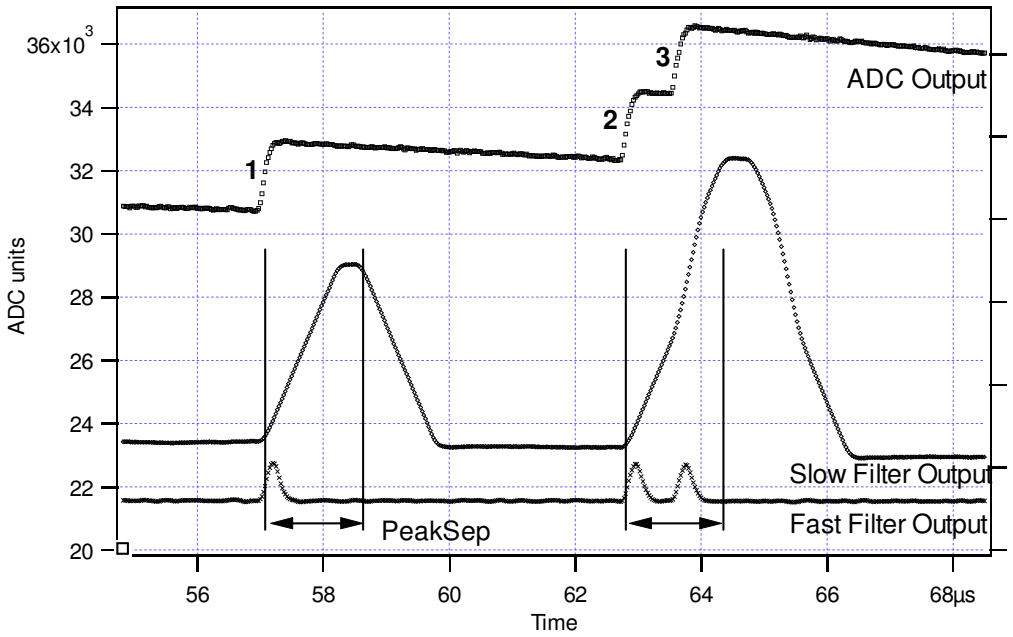

Figure 5.6: A sequence of 3 γ-ray pulses separated by various intervals to show the origin of pileup and demonstrate how it is detected by the Polaris.

The value V_x captured will only be a valid measure of the associated γ-ray's energy provided that the filtered pulse is sufficiently well separated in time from its preceding and succeeding neighbor pulses so that their peak amplitudes are not distorted by the action of the trapezoidal filter. That is, if the pulse is not *piled up*. The relevant issues may be understood by reference to Figure 5.6, which shows 3 γ-rays arriving separated by various intervals.

Because the triangular filter is a linear filter, its output for a series of pulses is the linear sum of its outputs for the individual members in the series. The fast filter has a rise time of 0.1 µs with a gap of 0.1 µs. The slow filter has a rise time of 1.2 µs with a gap of 0.35 µs.

Pileup occurs when the rising edge of one pulse lies under the peak (specifically the sampling point) of its neighbor. Thus, in Figure 5.6, peaks 1 and 2 are sufficiently well separated so that the leading edge of peak 2 falls after the peak of pulse 1. Because the trapezoidal filter function is symmetrical, this also means that pulse 1's trailing edge also does not fall under the peak of pulse 2. For this to be true, the two pulses must be separated by at least an interval of L + G. Peaks 2 and 3, which are separated by less than 1.0 µs, are thus seen to pileup in the present example with a 1.2 µs rise time.

This leads to an important point: whether pulses suffer slow pileup depends critically on the rise time of the filter being used. The amount of pileup which occurs at a given average signal rate will increase with longer rise times.

Because the fast filter rise time is only 0.1 µs, these γ-ray pulses do not pileup in the fast filter channel. The Polaris can therefore test for slow channel pileup by measuring the

© XIA LLC 2009. All rights reserved.

fast filter for the interval PEAKSEP after a pulse arrival time. If no second pulse occurs in this interval, then there is no trailing edge pileup. PEAKSEP is usually set to a value close to L + G + 1. Pulse 1 passes this test, as shown in Figure 5.6. Pulse 2, however, fails the PEAKSEP test because pulse 3 follows less than 1.0 μs. Notice, by the symmetry of the trapezoidal filter, if pulse 2 is rejected because of pulse 3, then pulse 3 is similarly rejected because of pulse 2.

5.5 Filter decimation

To accommodate the wide range of filter rise times from 0.1 μs to 44 μs, the filters are implemented in the RTPUs in FPGA configurations with different clock decimation (filter ranges). The ADC sampling rate is always 25ns, but in higher clock decimations, several ADC samples are averaged before entering the filtering logic. In decimation 1, 2^1 samples are averaged, 2^2 samples in decimation 2, and so on. Since the sum of rise time and flat top is limited to 31 decimated clock cycles, filter time granularity and filter time limits are as follows:

Decimation	Filter granularity	max. $T_{rise}+T_{flat}$	min. T_{rise}	Min. T_{flat}
1	0.05μs	1.55μs	0.15μs	0.15μs
2	0.1μs	3.1μs	0.5μs	0.3μs
3	0.2μs	6.2μs	0.4μs	0.6μs
4	0.4μs	12.4μs	0.8μs	1.2μs
5	0.8μs	24.8μs	1.6μs	2.4μs
6	1.6μs	49.6μs	3.2μs	4.8μs

Table 5.1**: RTPU clock decimations and filter time granularity**

As the decimations are implemented in different FPGA configurations, different files have to be downloaded to the FPGA to change decimation.

5.6 Count Rates and Livetime

During data acquisition, γ-rays will be absorbed in the detector at some rate. This is the *true input count rate*, which we will refer to as ICR_t. Because of pileup, not all of these will be detected by the Polaris' γ-ray pulse detection circuitry, which will thus report a *measured input count rate* ICR_m which will be less than ICR_t. This phenomenon, it should be noted, is a characteristic of all γ-ray detection circuits, whether analog or digital, and is not specific to the Polaris.

Of the detected γ-rays, some fraction will also satisfy pileup tests and have their values of V_x captured and placed into the spectrum. This number is the *output count rate*, which we refer to as the OCR. The standard description of extending dead time gives

© XIA LLC 2009. All rights reserved.

$$OCR = ICR * \exp(-ICR*DEADTIME) \quad (5.4)$$

Where, in the Polaris, DEADTIME is usually very close to the values 2*PEAKSEP, or, approximately 2*(L+G).

The Polaris normally returns, in addition to the collected spectrum, the actual time RUNTIME for which data was collected, together with the number FASTPEAKS of fast peaks detected and the number of V_x captured events NUMEVENTS. It also returns a quantity FTDT, the time the trigger filter spends above threshold and is not able to record further triggers.

From these values, both OCR and ICR_m can be computed according to Equation 5.5:

$$ICR_m = FASTPEAKS/(RUNTIME-FTDT)$$
$$OCR = NUMEVENTS/RUNTIME \quad (5.5)$$

The LIVETIME can then be computed as follows:

$$LIVETIME = RUNTIME* OCR/ ICR_m \quad (5.6)$$

In the Polaris Viewer, FTDT is used to compute a fractional Pileup Content. This parameter is defined as FTDT/ RUNTIME. The Pileup Content is thus an estimate of the pulses not counted in the FASTPEAKS due to pile up in the trigger.

Note that in the Polaris Viewer, runs are usually set to last for a given real time, which is different from RUNTIME. Real time is the laboratory time (Windows time) used to poll the Polaris; RUNTIME is the actual time the Polaris is ready to take data; not counting time for setting up the run and reading out the data. The difference is usually small.

© XIA LLC 2009. All rights reserved.

6 Appendix

This section contains hardware-related information.

6.1 Jumpers

6.1.1 Input (JP1, JP9 and JP10)

In the typical configuration, power for the detector preamplifier is supplied by the Polaris via the NIM standard DB-9 'Preamplifier Power' interface. The output signal and its reference return via BNC coaxial cable to the Polaris "input" connector. Some manufacturers instead route the signal back through the preamplifier power cable in order to save space. The Polaris accommodates either configuration using the "INPUT" jumpers JP9 and JP10. If set to "DSUB9", they connect pins 8 and 3 of the DB9 connector to the BNC shield and inner conductor, respectively. If set to "BNC", only the BNC connector is available for input signals. The default is "BNC".

JP1 connects the shield of the preamplifier power cable to board GND if set to the GND symbol. Otherwise, the cable shield is unconnected.

6.1.2 Signal Termination and Attenuation (JP108, JP109, JP112, JP113)

Near the input BNC connector are 4 jumpers to select the input impedance and attenuation. Jumper JP108 bypasses a 1kΩ attenuation resistor. Jumpers JP112, JP109 and JP113 connect to ground via 250Ω, 50Ω and 90Ω, respectively. If neither of jumpers JP112, JP109 or JP113 is set, the input is terminated with 10kΩ. Thus the jumper settings are as follows:

JP108	Remove only if you require attenuation. Attenuation will be 1: 5 if JP112 is set 1:21 if JP109 is set 1:12 if JP113 is set
JP112	Set for input impedance of 250Ω
JP109	Set for input impedance of 50Ω
JP113	Set for input impedance of 90Ω

Table 6.1**: Input impedance selection jumpers.**

6.1.3 Mode (JP103, JP104)

The analog signal conditioning circuitry can accommodate both resistive feedback and pulsed reset preamplifiers. JP103 and JP104 determine the digitally controlled signal to

© XIA LLC 2009. All rights reserved.

be subtracted from the preamplifier signal. In the default position labeled “RC”, a low-frequency offset, suitable for resistive feedback preamplifier types, is used. This digitally controlled offset voltage is sufficient to eliminate the average decaying sum of prior signal pulses in resistive feedback preamplifiers.

The pulsed reset preamplifiers are not supported by the current software.

6.1.4 VGA (JP106)

In normal operation, the signal gain is combined from a coarse gain set by relays and a digital fine gain applied in the DSP. An additional fine gain can be set by a variable gain amplifier (VGA). To do so, set JP106 to “on”. However, this should rarely be necessary, since the coarse gain is precise to about 2%. Furthermore, the VGA introduces additional electronic noise and thus is not recommended for optimal performance.

6.1.5 Compton Veto Polarity (JP110, JP111)

The “Gate” input accepts both positive and negative pulses from photomultiplier tubes. Set Jumpers JP110 and JP111 to “+” if the pulses are positive, to “-“ if they are negative.

6.1.6 HV shutdown (JP20, JP21)

Detectors with thermal shutdown protection usually come in two variants: Either logic high (one) indicates shutdown or logic low (zero) indicates shutdown. To match the Polaris’ shutdown circuitry to either variant, set JP20 to the logic level indicating the shutdown, and JP21 to the logic level indicating normal operation. This means that the Jumpers are always opposite to each other.

© XIA LLC 2009. All rights reserved.

6.2 Control and Status Register Bits

CSR, Modules	
0x0001	Run Enable
0x0002	New Run
0x0004	Unused, clear using CamacClear()
0x0008	LAM Enable
0x0010	DSP Reset
0x0020	FPGA Reset
0x0040	Unused, clear using CamacClear()
0x0080	Synch Flag
0x0100	FPGA Download Error in channel 0
0x0200	Reserved
0x0400	Reserved
0x0800	Reserved
0x1000	DSP Error
0x2000	Run Active
0x4000	LAM Request, cleared by reading the word count register
0x8000	Polaris Live

Table 6.2**: Map of the Control and status register (CSR).**

© XIA LLC 2009. All rights reserved.

6.3 Troubleshooting

6.3.1 IGOR reports “Function compilation error” at startup

- Check if the driver file Polaris.xop, is located in the “Igor Extension” folder (by default in C:\Program Files\Wavemetrics\IgorPro Folder). If not, copy the file from the *drivers* directory in the Polaris software distribution.

6.3.2 IGOR reports missing DLL file

- Make sure DLPortI/O is installed on your computer. It can be installed by running the program “port95nt.exe” located in the “drivers” subdirectory.

6.3.3 USB communication does not work

- Check if USB driver is installed.
- Verify that EPP port is not connected.
- Power cycle Polaris, give system enough time to initialize new USB device.

6.3.4 Igor reports missing files at system startup

- Check if the file names point to existing folders in your directory structure, especially if you installed the Polaris software in a custom directory.
- Check the filename of the settings file, make sure the file exists. You can create a new file by clicking on the *Create* button in the *System Configuration* tab.

6.3.5 Igor reports FPGA download unsuccessful at system startup

- Verify EPP cable or USB cable is connected properly, but not both.
- Verify Polaris is plugged in and switched on.
- If using EPP, verify that EPP or EPP+ECP was selected in your computer’s CMOS or BIOS setup. ECP is the abbreviation of Enhanced Capabilities Port. If only ECP was selected, the EPP connection for Polaris won’t work. Also verify the EPP address shown in the I/O panel matches the address on your computer. To do so, open the "System" Control Panel of MS Windows. In the Device Manager, look for "ports", and double click on the entry “printer ports” to display the port properties. The EPP address range is listed in the “Resources” tab.

6.3.6 Igor can not open files

- Make sure the directory structure is intake. In the main *Polaris* folder, there should be seven subfolders: *configuration*, *dsp*, *doc*, *drivers*, *firmware*, *MCA* and *pulseshape.*
- Make sure all files are not marked as read-only in the Windows operation system.

© XIA LLC 2009. All rights reserved.

6.3.7 No traces or only flat lines in Oscilloscope

- Click on “Adjust” to bring the DC offset in range.
- Check input signal connection.

6.3.8 Very high input count rate during run

- Check trigger threshold and increase above noise level.

6.3.9 Very low livetime during run

- Check trigger threshold and increase above noise level.

6.3.10 Large peak at low end of spectrum

- Check trigger threshold and increase above noise level.

6.3.11 Spectrum has very wide and blurred peaks

- Check and re-measure decay time.

6.3.12 Igor reports “need to have at least as many data points as fit parameters”

- This message occurs if an automatic fit in Igor failed. It can usually be ignored, but check the settings and maybe reload from file.

© XIA LLC 2009. All rights reserved.